

The Plotstuff Graphics Post Processor for Overture, User Guide, Version 1.00

W. D. Henshaw

April 20, 2000

U.S. Department of Energy

Lawrence
Livermore
National
Laboratory

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

This is a preprint of a paper intended for publication in a journal or proceedings. Since changes may be made before publication, this preprint is made available with the understanding that it will not be cited or reproduced without the permission of the author.

This work was performed under the auspices of the United States Department of Energy by the University of California, Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.

This report has been reproduced directly from the best available copy.

Available electronically at <http://www.doc.gov/bridge>

Available for a processing fee to U.S. Department of Energy
And its contractors in paper from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831-0062
Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-mail: reports@adonis.osti.gov

Available for the sale to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-mail: orders@ntis.fedworld.gov
Online ordering: <http://www.ntis.gov/ordering.htm>

OR

Lawrence Livermore National Laboratory
Technical Information Department's Digital Library
<http://www.llnl.gov/tid/Library.html>

The plotStuff Graphics Post Processor for Overture

User Guide, Version 1.00

Bill Henshaw

Centre for Applied Scientific Computing
Lawrence Livermore National Laboratory
Livermore, CA, 94551
henshaw@llnl.gov
<http://www.llnl.gov/casc/people/henshaw>
<http://www.llnl.gov/casc/Overture>

April 20, 2000

Abstract: This document explains how to use **plotStuff**, a program for plotting results from Overture programs. **plotStuff** reads data-base files created by Overture that contain information about the grid and solutions, a "show file". **plotStuff** has features for plotting contours, cutting planes, streamlines, iso-surfaces and grids. It can also be used to plot various derived quantities such as derivatives of the solution components. This document describes the class **DerivedFunctions** which is used to compute these derived quantities.

Contents

1 Introduction	2
2 Typical usage	2
3 The plotStuff menu items	3
4 Derived quantities	5
5 The DerivedFunctions class	6
5.1 constructors	6
5.2 getASolution	6
5.3 getASolution	6
5.4 getComponent	7
5.5 computeDerivedFunctions	7
5.6 add	7
5.7 remove	7
5.8 update	7

1 Introduction

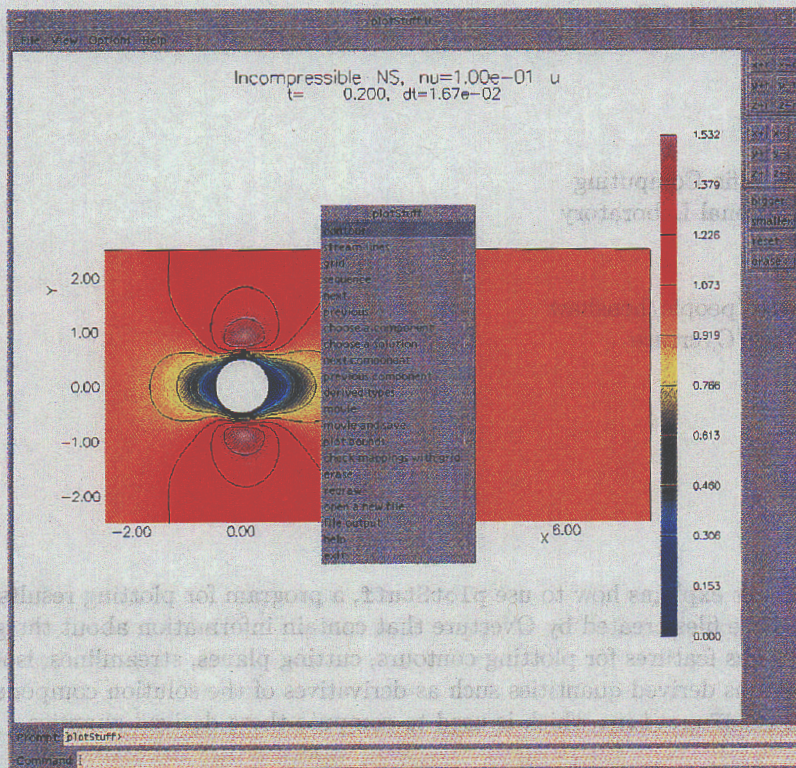


Figure 1: plotStuff is used to display results from an Overture data-base file ('show file') such as those created by the **OverBlown** flow solver

The plotStuff program, usually found in `Overture/bin/plotStuff` can be used to plot results found in "show files" or "grid files" created by Overture programs.

A "show file" contains a sequence of **frames**, each frame will normally contain a "solution" (i.e. a grid function defined on a grid). In the moving grid case each frame may also contain a new grid. Each solution will consist of a number of components, such as the velocity and pressure. A show file may also contain one or more **sequence**, which is an array of values such as the maximum residual over time.

Some of the features of plotStuff include

- plot contours of various flavours, plot streamlines, plot grids.
- show a movie of a time sequence of frames; save hard-copies for making mpeg movies.
- plot "derived types" from variables in the show file, such as plotting the vorticity given the velocity or plotting derivatives of the variables.

The plotStuff program is basically a front end to the features available in other classes such as `PlotStuff[2]`, `Ogshow[1]`, `ShowFileReader[1]` and the `DerivedFunctions` class.

2 Typical usage

The typical usage of plotStuff is to start the program up by typing "`plotStuff fileName.show`" or just "`plotStuff fileName`" since the ".show" is optional. Here `fileName.show` is the name of a show file such as one created by **OverBlown**[3]. Now one can choose contour to plot contours. At this point a new menu appears, this is the contour plotter menu. By choosing `exit` one returns to the main plotStuff menu. Choosing a new solution, or typing `next` or `previous` will plot contours of a different solution. One can also plot a different component. At this point one could choose `grid` or 'stream lines' to plot the grid or plot streamlines. Note that the contour plot will not be

erased by default. This allows multiple things to be plotted at once. One can choose **erase** to remove the contour plot before plotting other things. By choosing the '**derived types**' menu item one can build new components from existing ones. For example the vorticity or divergence can be created from a velocity vector. After these new derived types have been created they will appear in the **plotStuff** component menus.

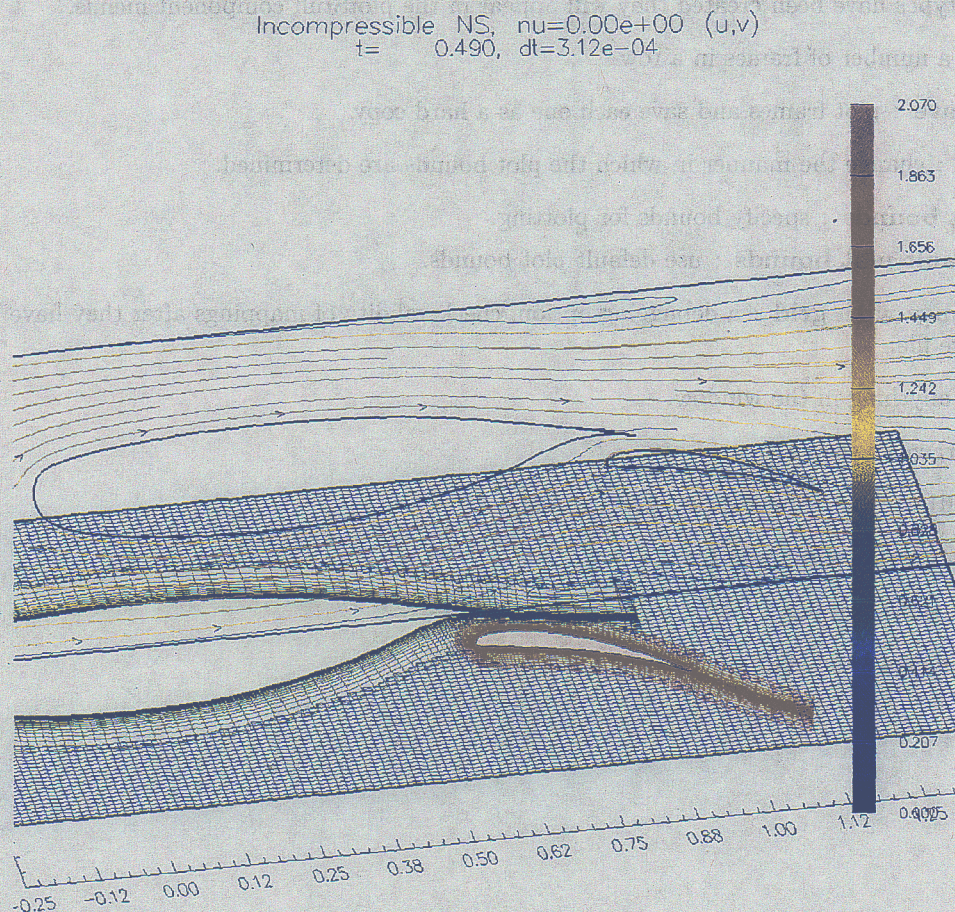


Figure 2: **plotStuff** can be used to plot more than one quantity on the same figure; here streamlines and a grid are plotted. This plot used an option in the grid-plotter to change the height of the grid. This computation was performed by the **OverBlown** solver.

3 The **plotStuff** menu items

Here is a description of the menu options available for **plotStuff**.

contour : plot contours.

stream lines : plot streamlines.

grid : plot the grid.

sequence : plot a sequence.

next : plot solutions from the next frame.

previous : plot solutions from the previous frame.

choose a component : plot a different component.

choose a solution : choose a different frame.

next component : plot the next component.

previous component : plot the previous component.

derived types : build new components as functions of the old ones, such as the vorticity from the velocity. One derived types have been created they will appear in the plotStuff component menus.

movie : plot a number of frames in a row.

movie and save : plot frames and save each one as a hard copy.

plot bounds : change the manner in which the plot bounds are determined.

set plot bounds : specify bounds for plotting.

use default plot bounds : use default plot bounds.

check mappings with grid : a debugging option, check validity of mappings after they have been read in from a data-base file.

erase : erase anything in the window.

redraw : redraw the screen.

open a new file : open a new show file for reading.

file output : output results to a file.

help : print a short help list.

exit : exit this menu and continue on (same as 'continue').

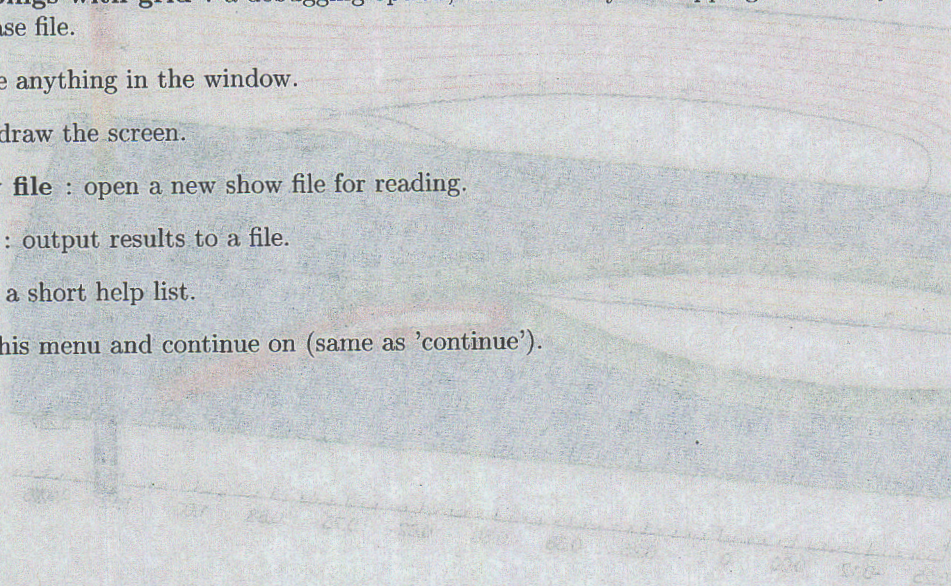


Figure 2: plotStuff can be used to plot more than one quantity on the same figure, here streamlines and a grid. The plot used an option in the grid-plotter to change the height of the grid. This computation was performed by the OverFlow solver.

3 The plotStuff menu items

Here is a description of the menu options available for plotStuff.

contours : plot contours

streamlines : plot streamlines

grid : plot the grid

sequence : plot a sequence

next : plot solutions from the next frame

previous : plot solutions from the previous frame

choose a component : plot a different component

4 Derived quantities

Various derived quantities can be plotted with `plotStuff`. The class `DerivedFunctions`, described in more detail in the next section, is used to define derived functions for plotting with `plotStuff`. For example one can compute the vorticity given some velocity components. One can also differentiate components.

Here are some of the derived functions we can create

x,y,z,xx,yy,zz,laplacian : compute first and second derivatives of any components,

$$x = \partial_x u$$

$$y = \partial_y u$$

$$z = \partial_z u$$

$$xx = \partial_{xx} u$$

$$yy = \partial_{yy} u$$

$$zz = \partial_{zz} u$$

$$\text{laplacian} = \Delta u = \nabla \cdot \nabla u = \partial_x^2 u + \partial_y^2 u + \partial_z^2 u$$

vorticity The same as `zVorticity`, see below. In 2D this is the only non-trivial component of the vorticity.

xVorticity, yVorticity, zVorticity Components of the curl of the velocity, $\nabla \times \mathbf{u}$,

$$\text{xVorticity} = \partial_z v - \partial_y w$$

$$\text{yVorticity} = \partial_x w - \partial_z u$$

$$\text{zVorticity} = \partial_y u - \partial_x v$$

divergence The dilatation or divergence of the velocity, $\nabla \cdot \mathbf{u}$,

$$\text{divergence} = \partial_x u + \partial_y v + \partial_z w$$

mach Number The ratio of the local magnitude of velocity to the local speed of sound:

$$\text{mach Number} = \frac{\sqrt{u^2 + v^2 + w^2}}{a}$$

$$a = \sqrt{\gamma R_g T} \quad \text{if } T \text{ is defined}$$

$$a = \sqrt{\gamma p / \rho} \quad \text{if } p \text{ and } \rho \text{ are defined}$$

temperature The temperature as a function of the pressure, density and γ ,

$$\text{temperature} = \frac{\gamma p}{\rho}$$

speed The magnitude of the velocity, $\sqrt{\mathbf{u} \cdot \mathbf{u}}$,

$$\text{speed} = \sqrt{u^2 + v^2 + w^2}$$

minimumScale, r1MinimumScale, r2MinimumScale, r3MinimumScale Local measures of the minimum scale...

The components u, v, w etc. are determined by looking for the following variables in the show file,

uComponent : the component number of x-component of the velocity u .

vComponent : the component number of y-component of the velocity v .

wComponent : the component number of z-component of the velocity w .

pressureComponent : the component number of the pressure p .

temperatureComponent : the component number of the temperature T .

densityComponent : the component number of the density ρ .

For example, **OverBlown** saves the incompressible flow components with **pressureComponent**=0, **uComponent**=1, **vComponent**=2, and **wComponent**=3. See the **OverBlown** file **saveShow.C** for an example of saving this extra info in the show file.

Other variables that may be required in the show file are

gamma : ration of specific heats (if constant).

Rg : the gas constant.

nu : kinematic viscosity.

5 The DerivedFunctions class

Here is a description of the **DerivedFunctions** class.

5.1 constructors

DerivedFunctions(ShowFileReader & showFileReader_)

Description: Derive new functions from old ones. This class is used by **plotStuff** to create derived quantities such as vorticity, mach-number, derivatives etc.

5.2 getASolution

```
int  
getASolution(int & solutionNumber,  
             MappedGrid & cg,  
             realMappedGridFunction & u)
```

5.3 getASolution

```
int  
getASolution(int & solutionNumber,  
             CompositeGrid & cg,  
             realCompositeGridFunction & u)
```

5.4 GetComponent

```
int  
GetComponent( int & c, const String & cName )
```

Access: protected.

Description: Look for a component number in the data base.

c (input): c==-2 first time through. If not found c==-1 on output.

Return value: 0=success, 1=not found.

5.5 computeDerivedFunctions

```
int  
computeDerivedFunctions( realCompositeGridFunction & u )
```

Access: protected.

Description: This function will compute the derived types and add these as new components to the end of the grid function u. This function is normally called after reading a new solution from the show file. It used the showFileReader pointer to get info about the current frame.

u (input/output): On input u should contain valid values for the default components. On output u will be redimensioned to hold the extra derived components.

5.6 add

```
int  
add(int derivative, const String & name_,  
    int n1 =0,  
    int n2 =0)
```

Description: Add a new entry to the list of derived functions.

derivative (input) : derived type code.

name_ (input) : component name for this entry.

n1,n2 (input) : extra info to save in the derived array.

5.7 remove

```
int  
remove( int i )
```

Description: remove an entry from the list of derived types.

5.8 update

```
int  
update(GenericGraphicsInterface & gi,  
       int numberOfComponents,  
       String *componentNames )
```

Description: update current list of derived grid functions

References

- [1] W. HENSHAW, *Ogshow: Overlapping grid show file class, saving solutions to be displayed with plotStuff, user guide*, Research Report UCRL-MA-132235, Lawrence Livermore National Laboratory, 1998.
- [2] —, *Plotstuff: A class for plotting stuff from Overture*, Research Report UCRL-MA-132238, Lawrence Livermore National Laboratory, 1998.
- [3] —, *OverBlown: A fluid flow solver for overlapping grids, user guide*, Research Report UCRL-MA-134288, Lawrence Livermore National Laboratory, 1999.

Index

- class DerivedFunctions, 6
- derived types, 3
- file output, 4
- graphics post processing, 2
- movie, 4
- plot bounds, 4
- plotStuff, 2
- plotting derived quantities, 5
- show files, 2
 - plotting, 3